

REMARKS

The above amendments and following remarks are submitted as a Preliminary Amendment on even date accompanying a Request for Continued Examination in response to the pending Final Official Action of the Examiner mailed October 17, 2008¹. Having addressed all objections and grounds of rejection, claims 1-21, being all the pending claims, are now deemed in condition for allowance. Entry of this amendment and reconsideration to that end is respectfully requested.

Claims 1, 6, 11-13 and 16-19 have been rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 5,574,898, issued to Leblang et al (hereinafter referred to as "Leblang"). The ground of rejection is respectfully traversed for the following reasons.

The standards for a finding of anticipation during examination are specified in MPEP 2131, which provides in part:

TO ANTICIPATE A CLAIM, THE REFERENCE MUST TEACH
EVERY ELEMENT OF THE CLAIM

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). **"The identical invention must be shown in as complete detail as is contained in the ... claim."** *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). (emphasis added)

¹ The pending Official Action is *prima facie* premature because it contains new grounds of rejection for unamended claims (see for example claim 21).

The rejection is respectfully traversed because "the identical invention" is not shown by Leblang "in as complete detail as is contained in the claims" as is required by MPEP 2131.

Applicants' invention as disclosed and claimed is a data processing system for efficiently accessing shared JavaScript objects within a multi-user environment by reducing the overhead required to maintain object/database property synchronization. A version is associated with each object to indicate whether any particular requested access is associated with the most current properties of the object. Each user has a version list which indicates whether any particular object properties need be updated before access. The version list is maintained in faster and more available memory than the actual object to enhance efficiency. The users of Applicants' invention are thus involved in and responsible for the maintenance of a data base.

In the preferred mode, an additional property is described in the physical data base table and corresponding property in the JavaScript table object. This property is referred to as the version property. The purpose of this property is to track the number of modifications made to its associated data base table.

The essential element is that JavaScript object implementation code be able to quickly compare the current version property value of the data base table to the version property value stored in the JavaScript object the last time the

physical data base table properties were retrieved. If the values are the same, no refresh of the object's properties is required. If the values differ, they must be refreshed before proceeding with any table reference.

Unlike Applicants' invention which is dedicated to "Synchronizing Dataset Object Properties" (see title), Leblang is concerned with controlling "Dynamic Software Version" (see title). In other words, Applicants' claimed system is directed to data change maintenance whereas Leblang is directed to software program maintenance. Though both are associated with "version" control of "objects", Applicants control "data objects", whereas Leblang controls "software program objects". Even though both types of "objects" can be expressed in binary terms, the problems to be solved and the systems impacts of proper/improper version maintenance are so disparate that the structures employed by Applicants and Leblang are radically different.

For example, in Applicants' system, the user far more responsible for use of the appropriate (dataset) version, whereas Leblang assigns most of the responsibility for software version maintenance to a centralized "auditor". This difference is to be expected, because the users of Applicants' disclosed and claimed system operate upon the data within an Enterprise database, whereas by way of contrast, the Leblang disclosure is a Computer-

Aided Software Engineering (CASE) system. Column 1, lines 63-64, of Leblang states:

It is the object of the present invention to provide a CASE system....

The structural differences are most apparent as the individual claim limitations are compared with Leblang. Particular attention should be directed to the nature, location, and use of the "object"; the nature of the "version property"; and the structure and operation of the comparison facility.

Apparently, the Examiner is unable to appreciate these distinctions. Therefore, independent claims 1, 6, 11, 16, and 21 have been amended to make these distinctions even more explicit. For example, the claims have been amended to ensure that the interaction of the user with the data base are to "manipulate" (i.e., move and change) the data, as summarized in the specification at page 10, lines 3-5. This should be readily distinguishable from the executable software disclosed by Leblang.

Similarly, the claims have been amended to show that the claimed "data set objects" are associated with "physical data base tables", as supported in the specification at page 11, lines 16-17. In addition, the claims have been amended to show that the claimed "version list" is stored within a faster memory than that utilized for storage of the basic data base. This feature is highlighted in the specification at page 33, lines 8-9. The

impact of these changes is to more explicitly distinguish from the prior art of record as shown below in detail.

Claim 1, as amended, has seven basic elements. The first element is "an enterprise server containing a data base having a plurality of data objects each representing a physical data base table". Instead of addressing this element, the Examiner improperly parses the claim element into a plurality of largely unrelated pieces. As such, the Examiner has found that "servers" 106 are equivalent to the claimed "enterprise server". This finding is clearly erroneous, because "servers" 106 merely provide communication for software developers to Version Object Bases (VOB) 102 within a distributed environment.

Yet, the Examiner clearly erroneously finds that public storage device 20 is the claimed "data base" of "server" 106. This is contrary to the teaching of Fig. 1 of Leblang, which shows no coupling between servers 106 (of Fig. 2) with public storage device 20 (of Fig. 1). In fact, Figs. 1 and 2 show mutually exclusive systems, wherein Fig. 2 is the "distributed" embodiment.

Finally, the Examiner clearly erroneously finds that "VOB's", which are object language descriptions of software programs or portions thereof are somehow the claimed "data objects". Again, even though VOB's are conceivably expressed in binary terms, they are clearly not the claimed "data objects" of

the claimed "database" of the claimed "enterprise server". Furthermore, they do not represent "a physical data base table", as amended in the claim.

The newly added second claimed element is "a mass storage system having a first access time and a memory having a second access time wherein said second access time is faster than said first access time". There is no showing of different classes of memory within Leblang.

The third claimed element is "a version property associated with one of said plurality of data objects". Applicants, being their own lexicographers as a matter of law, define "version property" throughout their disclosure. However, a convenient summary is found at page 12, line 22, through page 13, line 3, which states:

In the preferred mode, an additional property is described in the physical data base table and corresponding JavaScript table object. This property is referred to as the Version property. The purpose of this property is to track the number of modifications made to its associated data base table.

Again, because Leblang does not have the claimed element using Applicants' definition, the Examiner impermissibly attempts to fabricate it using Leblang's definition. In doing so, the Examiner column 18, lines 37-41, which is not the claimed "version property" and is not "associated with one of said plurality of data objects" as defined and claimed by Applicants.

The fourth claimed element is "a terminal having a session which generates a request under control of one of said plurality of said users involving access to said one of said plurality of data objects and permit manipulation of said data". Because Leblang does not disclose these limitations, the Examiner simply ignores the claimed "session", for example. Furthermore, as explained above, Leblang does not have the claimed "data objects", because it is a software development tool.

Taking exception with Applicants' position, the Examiner cites Fig. 23 and column 2, lines 57-62 and Leblang. Not only are neither of these two citations related to the fourth claimed element, they are not even related to each other.

The fifth claimed element is "a version list stored within said memory associated with said session having an assumed version property". Because Leblang does not have the claimed "session" or the claimed "version property", it cannot have the claimed "version list" associated therewith. Thus, the Examiner again attempts to paraphrase the limitation and cites unrelated and irrelevant material which does not address Applicants' invention. As explained above, Leblang also does not disclose the claimed different classes of memory.

Leblang does not have the claimed "version property" or "version list". Therefore, it cannot have the claimed "comparing means" of claim element six or the "update facility" of claim

element seven. As a result, the Examiner again makes a plurality of unrelated and disparate citations. For example, the Examiner cites the Abstract of Leblang to establish:

An audit record comparator is provided for determining the difference between source object versions used in building two or more derived object versions.

It is not understood why one would consider the disclosed "audit comparator" for comparing software source code is related to the claimed "comparing means" for comparing "version properties". Applicants' compare variables (i.e., "version properties") which describe the object data, whereas Leblang compare the object data itself. Surely, one can understand this distinction.

Even more confusing is that the Examiner attempts to equate "updating" of a software program in Leblang with the claimed "updating" of the "version list". Leblang "updates" the object program, whereas Applicants update a description of the object data.

As a result of Leblang having none of the seven claimed elements of amended claim 1 and virtually none of the irrelevantly paraphrased elements, the rejection of amended claim 1, and all claims depending therefrom, is respectfully traversed.

Claim 6, as amended, is an independent method claim having five steps as limiting elements. As explained above, Leblang is simply not pertinent to the claimed method, because it does not have the claimed "data set object....representing a physical data

base table" or employ the claimed "version property" or the claimed "version list" to accomplish database synchronization. Instead, Leblang provides a software development tool. As a result, Leblang cannot perform the steps of the claimed invention and therefore, the Examiner provides clearly erroneous findings of fact and clear errors of law, for the purpose of "manufacturing" an alleged case of anticipation.

For example, the Examiner again makes the clearly erroneous and legally incorrect assertion that "attributes associates (sic) with each version" are equivalent to the claimed "version property". In addition, the Examiner makes legally irrelevant findings with regard to the claimed "preparing" step, because Leblang has no "version list" as claimed and no "version property" as claimed. Because it does not disclose different classes of memory, it cannot disclose the storage arrangement of the claimed "version list".

Furthermore, the Examiner again clearly erroneously alleges that comparing different versions of software source code programs is somehow the same as comparing "version properties". Finally, because Leblang has no "version list" as claimed, the Examiner clearly erroneously alleges that updating the software programs is the same as updating the claimed "version property".

Thus, having none of the claimed steps which limit amended claim 6, the rejection of amended claim 6, and all claims depending therefrom, is respectfully traversed.

Claim 11, as amended, contain "means-plus-function" limitations which must be examined in accordance with MPEP 2181-2184. Clearly, this has not been done. For example, MPEP 2181 requires that the Examiner must expressly acknowledge these "means-plus-function" limitations. In fact, claim 11 has not been examined at all. Instead the Examiner states:

With respect to claim 11, Leblang et al discloses the same limitations as recited on claims 6 (see rejection on claim 6).

This statement is incorrect as a matter of law, because claim 11 contains "means-plus-function" limitations and claim 6 does not. It is also clearly erroneous as a matter of fact, because claim 6 contains method steps as limitations and claim 11 contains apparatus elements as limitations.

Therefore, the rejection of amended claim 11, and all claims depending therefrom, is respectfully traversed in view of the numerous errors discussed above and for failure to be examined in accordance with controlling law.

Claim 12 depends from claim 11 and is further limited by "updating means responsively coupled to said comparing means for updating said version list if said comparing means finds said version property different from said assumed version property".

Because Leblang does not have the claimed "version list", the claimed "version property" or the claimed "assumed version property". Therefore, the Examiner ignores these limitations and cites material from Leblang associated with updating a software program rather than the claimed "version list". The rejection of claim 12 is respectfully traversed.

Claim 13 depends from claim 12 and is further limited by "a publically accessible digital data communication network which couples said requesting means to said storing means". Leblang certainly does not have this network. Therefore, the Examiner cites Fig. 1, showing a single computer, and Fig. 2, showing a local area network (LAN). Clearly, Figs. 1-2 do not contain the claimed coupling network. Therefore, the Examiner makes a largely incoherent argument having nothing to do with coupling networks or Figs 1-2. The rejection of claim 13 is respectfully traversed.

Amended claim 16 is an independent apparatus claim having three major limitations. Instead of actually examining claim 16, the Examiner states:

With respect to claim 16, Leblang et al discloses the same limitation (sic) as recited on (sic) claim 1 (see rejection of claim 1).

A cursory review of claims 1 and 16 will show that this statement is clearly erroneous. Furthermore, the statement is incorrect as a matter of law. For example, claim 16 is explicitly limited to

location of the claimed "version property". Therefore, the rejection of amended claim 16, and all claims depending therefrom, is respectfully traversed in view of the numerous errors discussed above and for failure to be examined as required by controlling law.

Claim 17 depends from claim 16 and is further limited by "update facility responsively coupled to said comparison facility and said version list which updates said version list if said comparison facility finds said version property different from said assumed version facility". Leblang does not have the claimed "version list", the claimed "version property", nor the claimed "assumed version property". Therefore, the Examiner ignores Applicants' claimed invention and cites Leblang, column 34, lines 25-27, to irrelevantly show that Leblang updates the software under development. The rejection of claim 17 is respectfully traversed.

Claim 18 depends from claim 17 and is further limited by "wherein said session is responsively coupled to said data base management system via a publically accessible digital data communication network". Leblang has no "session" and has no "network" as claimed. Therefore, the Examiner supports his rejection by citing material from Leblang having nothing to do with the claimed "session" or the claimed "network". Furthermore, the Examiner apparently contradicts himself by

suggesting that claim 18 is rejected for the same reasons as claim 2, for which the Examiner explicitly admits that Leblang does not anticipate claim 2. The rejection of claim 18 is respectfully traversed as admittedly improper.

Claim 19 depends from claim 18 and is further limited by "wherein said version list is stored in a memory having a faster access time than a memory containing said dataset". Thus, claim 19 depends from admittedly improperly rejected claim 18.

In addition, Leblang does not have the claimed "version list". Furthermore, though the claim requires the "version list" to be stored in faster memory than the dataset, the Examiner nevertheless cites Leblang; column 10, lines 61-62, which irrelevantly discusses storage of a version of the software under development. The rejection of claim 19 is respectfully traversed.

Claims 2-5, 7-10, and 20-21 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Leblang in view of U.S. Patent No. 7,100,195, issued to Underwood. This ground of rejection is respectfully traversed for failure of the Examiner to present a *prima facie* case of obviousness as specified by MPEP 2143.

To make a *prima facie* case of obviousness, MPEP 2143 requires the Examiner to provide evidence and argument showing: 1) motivation to make the alleged combination; 2) reasonable

likelihood of success of the alleged combination; and 3) all claimed elements within the alleged combination. The Examiner has failed to make any of these three required showings. Therefore, because the Examiner has not made a *prima facie* case of obviousness, Applicants need not and indeed cannot offer appropriate evidence and argument in rebuttal.

In an apparent attempt to allege motivation, the Examiner concludes:

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify Leblang et al.'s system by using JavaScript in order to create the good environment to provides (sic) a level of interaction fast (sic) and more complex between the clients and server and the useful and powerful of (sic) JavaScript language on its web server for the stated purpose that has been well known in the art as evidenced by teaching of Underwood (col. 1, lines 60-68). (emphasis added)

This statement is largely incomprehensible. However, to the extent understood, the Examiner seems to conclude that the alleged combination would "create the good environment" without any evidence that the environment of Leblang is otherwise not a good environment. Furthermore, totally without evidence, the Examiner appears to conclude that the alleged combination would make Leblang "fast and more complex" without any evidence that Leblang is slow and less complex.

However, most glaringly erroneous is that the Examiner concludes that the alleged combination with Underwood would enhance the "web server" of Leblang, even though Leblang does not

have a "web server". In fact there is no showing that Leblang is even coupled to the web, because it is a software development tool used in a private and proprietary environment. Coupling Leblang to the Internet would appear to be neither necessary nor desirable.

Having failed to show any motivation to make the alleged combination, the Examiner simply ignores the required showing of reasonable likelihood of success. Most likely this results from the readily apparent incompatibilities between Leblang and Underwood. The third required showing is that of all claimed elements within the alleged combination. This is best viewed with reference to the actual claim limitations.

Claim 2 depends from claim 1 and is further limited by "wherein said user session further comprises a JavaScript object". As explained above, Leblang has no "user session" as claimed. And as admitted by the Examiner Leblang has no "JavaScript object" as claimed. Therefore, the Examiner improperly parses the term "JavaScript object" into "Script object" and "Java". He then alleges that Leblang discloses "script object" and Underwood discloses "JavaScript". Thus, these findings are inadequate, incomplete, clearly erroneous, and contrary to controlling law. The rejection of claim 2 is respectfully traversed.

Claim 3 depends from claim 2 and is further limited by "wherein each of said plurality of data objects has a separate version property associated therewith". Leblang has no "plurality of data objects" and the Examiner has not even alleged that it does. As explained above, Leblang has no "version property". Therefore, Leblang cannot have the further limitations of claim 3. The rejection of claim 3 is respectfully traversed.

Claim 4 depends from claim 3 and further limits the network for coupling the claimed "user" or "requesting means" to the claimed "data base". Because Leblang has neither the claimed "user" or the "data base", it cannot possibly meets the further limitations of claim 4. The rejection of claim 4 is respectfully traversed.

Claim 5 depends from claim 4 and is further limited by "wherein each of said version properties is stored within said data base". Leblang does not have the claimed "data base" or the claimed "version property" and the Examiner does not allege that it does. Therefore, Leblang cannot have the further limitations of claim 5. The rejection of claim 5 is respectfully traversed.

Claim 7 depends from claim 6 and is further limited by "wherein said user session further comprises a JavaScript object". The alleged combination has no "JavaScript object" as claimed. Certainly, the alleged combination does not have the

claimed "session" as explained above. The rejection of claim 7 is respectfully traversed.

Claim 8 depends from claim 7 and further limits the memory in which the claimed "version list" is stored. Because the alleged combination does not disclose the claimed "version list" the Examiner makes a number of findings which are largely incoherent, unsupported by the prior art of record, clearly erroneous, and legally irrelevant. In fact, the Examiner's citation establishes that the software program being developed is stored in memories of differing speeds, but says nothing about storing the claimed "version list". The rejection of claim 8 is respectfully traversed.

Claim 9 depends from claim 8 and further limits the coupling network. As explained above, the alleged combination cannot meet the limitations of claim 8 from which claim 9 depends. Therefore, the alleged combination cannot meet the further limitations of claim 9. Furthermore, the Examiner appears to agree that the alleged combination does not meet the limitations of claim 9, so he seeks to apply something he calls "Avery et al.", which is not a portion of the alleged combination used to reject claim 9. The rejection of claim 9 is respectfully traversed.

Claim 10 depends from claim 9 and further limits the "version property". The alleged combination does not have the

claimed "version property" as discussed above. Therefore the alleged combination cannot meet the limitations of claim 10. Furthermore, the Examiner appears to agree that the alleged combination does not meet the limitations of claim 10, so he seeks to apply something he calls "Avery et al.", which is not a portion of the alleged combination used to reject claim 10. Thus, the rejection of claim 10 is respectfully traversed.

Claim 18 has been rejected by the Examiner as anticipated by Leblang on page 6 of the pending final official action. He apparently contradicts himself by suggesting that claim 18 is rejected for the same reasons as claim 2, for which the Examiner explicitly admits that Leblang does not anticipate claim 2. The rejection of claim 18 is respectfully traversed as admittedly improper.

Claim 20 depends from claim 19 and is further limited by "wherein said object further comprises a JavaScript object". The Examiner does not actually examine claim 20 but refers to the rejection of claim 2, which was traversed as being impermissibly parsed so that the Examiner can find all of the words of the claim somewhere within the alleged combination. The rejection of claim 20 is respectfully traversed.

Amended claim 21 is an independent apparatus claim having six limiting elements. Instead of actually examining claim 21 as required by controlling law, the Examiner states:

With respect to claim 21, Leblang et al. discloses the same limitation (sic) as recited on (sic) claim 1 (see rejection on claim 1) and Underwood discloses the Javascript with the same motivation of claim 2.

In addition to being largely incoherent, this statement is clearly erroneous and erroneous as a matter of law. Furthermore, the rejection is improper as admitted by the Examiner. The rejection of claim 21 is respectfully traversed.

Claims 14-15 have been rejected as unpatentable over Leblang in view of Underwood and further in view of U.S. Patent No. 5,917,485, issued to Spellman et al (hereinafter referred to as "Spellman". This ground of rejection is respectfully traversed for failure of the Examiner to present a *prima facie* case of obviousness as specified by MPEP 2143.

As to the requirement to show motivation, the Examiner concludes:

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify Leblang et al.'s system by using the Mapper data base management system in order to have data base management system in an efficient, multi-user environment and to enable the user to utilize either access technique, the logic for each individual assistance function for the stated purpose has been well known in the art as evidenced by teaching of Spellman (col. 2, line 28-38).

However, Leblang has no data base management system at all.

Furthermore, the Examiner does not allege that it does. As explained above, Leblang is a software development tool.

Therefore, there can be no motivation *per se* for improving the

efficiency of Leblang's non-existent data base management system by making it the claimed MAPPER system.

The Examiner does not venture any showing of reasonable likelihood of success as required by MPEP 2143. However, he could not do so, because of the readily apparent incompatibilities of Spellman, Underwood, and Leblang.

Finally, the Examiner fails to show all of the claimed elements. In making his findings, the Examiner clearly erroneously states:

Leblang at al. discloses wherein said requesting means further comprising (sic) an industry standard personal computer (fig. 2).

None of the alleged devices disclosed by Fig. 2 of Leblang meets the requirement to show the "identical invention in as complete detail as found in the claim". The rejection of claims 14-15 is respectfully traversed for failure of the Examiner to make any of the three showings required by MPEP 2143.

Having thus responded to each objection and ground of rejection, Applicants respectfully request entry of this amendment and allowance of claims 1-21 being the only pending claims.

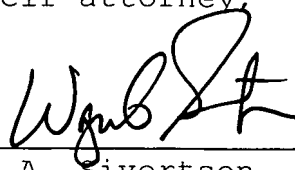
Please charge any deficiencies or credit any overpayment to
Deposit Account No. 14-0620.

Respectfully submitted,

Barbara A. Christensen et al.

By their attorney,

Date January 19, 2009



Wayne A. Sivertson
Reg. No. 25,645
Suite 401
Broadway Place East
3433 Broadway Street N.E.
Minneapolis, Minnesota
55413
(612) 331-1464